i) logically dividing area of the image into a coarse Z-buffer, the coarse Z-buffer including a series of tiles, the tiles being arranged in a rectangular grid, wherein the grid may have different resolutions, and wherein each tile has an associated depth value, the depth value being a Z-buffer value farthest from the eye that is included within that tile;

ii) constructing a surrogate volume for each object of the image, wherein each surrogate volume is a three-dimensional object that is just large enough to contain the object being ordered and wherein each surrogate volume may span only one tile of an appropriate resolution;

B1 Cont

iii) determining a depth value of the surrogate volume that is nearest to eye of a viewer;

iv) determining a depth value of the one tile that includes the surrogate volume;

a host processor for:

determining which objects are occluded by comparing the depth value of the surrogate volume versus the depth value of the tile including the surrogate volume,

generating the surrogate volumes for the objects being processed, and

transforming the surrogate volumes from object space to eye space; and

a graphics processor for rendering the objects that are not occluded.

REMARKS

Claims 1-13 were pending in the above-identified application when last examined. Claims 1-13 were rejected by the Examiner. Applicant respectfully traverses the Examiner's obviousness rejections as to claims 1-13. New claims 14-18 are being added. No new matter is being added by virtue of this amendment.

Claim Rejections Under 35 U.S.C. § 103:

The Examiner rejected claims 1-13 under 35 U.S.C. 103(a) as unpatentable over *Aleksicy*, US Patent 5,977, 980, in view of *Quaknine et al.*, US Patent 6,091,422. Applicant respectfully traverses the rejection.

In section 4 of the Office Action, the Examiner rejects claims 1-10 under the same verbatim rationale presented in the previous office action of February 5, 2001. The Examiner further states that added claims 11-13 are based on the method of claim 1 and rejected under a similar rationale.

In section 5 of the Office Action, the Examiner responds to the arguments presented by the Applicant in the response to the previous office action. The Examiner states:

> Applicant argues that (a) the cited references do not teach or suggest dividing the z-buffer into a series of tiles (b) a surrogate volume; [or that] (c) each tile in the lower resolution coarse z-buffer covers the same screen area as each tile in the coarse z-buffer.
>
> Per (a) *Aleksicy*'s computer display being divided into a plurality of tiles does suggest the same idea of dividing the Z-buffer into a series of tiles (col. 2, lines 25-31, and fig. 1). Furthermore *Quaknine*'s the render region is broken into tiles teaches that the step of dividing Z-buffer into a series of tiles is widely used in the art (*Quaknine*, col. 5, lines 65-66).
>
> Per (b) *Aleksicy* teaches surrogate volume in column 3, lines 25-26.
>
> Per (c) *Quaknine* teaches each tile in the lower resolution coarse Z-buffer covers the same screen area as each tile in the coarse Z-buffer in column 8, lines 48-64.

The passages of *Aleksicy* and *Quaknine* cited above by the Exmainer follow:

> Generally, the present invention provides a method and apparatus for determining visibility of a pixel during video rendering. This may be accomplished by determining z-positioning information of an object element. The z-positioning information is representative of the z-information of an object element in a particular section of the display.

Aleksicy, column 2, lines 25-31.

> In this particular embodiment, the view of which the render region is a part, is broken into tiles.

Quaknine, column 5, lines 65-66.

> To render the three images in tile 18, each object element of each object is individually rendered.

Aleksicy, column 3, lines 25-26.

> Another alternative to limiting the computational demands of the rendering is to allow the user to adjust the resolution of the realistic rendering. In a variation, the realistic rendering is subdivided into portions, selected portions being rendered sequentially by the second synchronous thread. The selected portions are those that are

determined to change appearance as a result of the update data. This also reduces the computational demand by not reproducing rendered portions. In another variation, the second thread is interrupted when the update data is received and reinstated after the update data becomes available to the rendering process of the second thread. This insures that data, invalidated by update data, is not used in the rendering process. Invalidated portions may be blanked in a display prior to re-rendering so stale parts of the rendering are never displayed. Alternatively, the stale portions remain the display until overwritten by the updated portions.

Quaknine, column 8, lines 48-64.

Applicant respectfully disagrees that the structure of the Z-buffer, surrogate volume, or each tile of the lower resolution coarse Z-buffer covering the same area as each tile in the coarse Z-buffer are disclosed or suggested by the passages pointed out by the Examiner or elsewhere in *Aleksicy* or *Quaknine*, considered alone or in combination.

In response to Examiner's argument in (a) that dividing the Z-buffer into tiles is prior art, Applicant respectfully requests reconsideration in view of the following remarks:

As the Examiner correctly observes, *Aleksicy* does not disclose "constructing a coarse Z-buffer, the coarse Z-buffer subdivided into a series of tiles". Further, contrary to the Examiner's assertion, coloumn 9, lines 63-66 of *Quaknine* state: "The realistic rendering is displayable in a resizable window that can be adjusted to alter a number of pixels contained therein and thereby limit the computational demands of maintaining a realistic rendering." This passage is talking about a resizable window. It does not teach or suggest dividing the Z-buffer into a series of tiles; nor does it suggest that such practice is widely used in the art. On the contrary, *Quaknine* discloses the approach of eliminating redundant pixel fragment operations and teaches away from constructing a coarse Z-buffer. Neither do the combination of *Aleksicy*, column 2, lines 25-31 and figure 1 and *Quaknine* column 5, lines 65-66 teach or suggest dividing the Z-buffer into tiles. Moreover, when it comes to culling objects, *Aleksicy*'s focus again is on pixels and its algorithm discards the pixel information, and not the entire object, for only those portions of the object that are overlapped.

Neither reference discloses or suggests a Z-buffer divided into tiles, and nor does the combination of the two. Moreover, as *Aleksicy* is focused on comparing the z parameters of the object elements and *Quaknine* is focused on the comparison of the z parameters of pixels, there is no suggestion in either reference to combine the two methods that are mutually exclusive.

In response to Examiner's argument in (b) that *Aleksicy* teaches the surrogate volume in column 3, lines 25-26, please note the following:

Aleksicy teaches individually rendering each object element and this teaches away from the tile approach and the surrogate volume approach. Further, *Quakenine* does not include any discussion of a surrogate volume. The approach of *Aleksicy* is qualitatively different from the comparison of surrogate volumes. Surrogate volumes are different from the object elements of *Aleksicy*; each surrogate volume corresponds to one tile and not more than one tile; and, therefore, only one comparison between the z-values of the surrogate volume and the tile is required per tile. *Aleksicy* compares the z-values of the object elements within each tile with the z value of the tile and this process usually involves more than one comparison; further, if this approach fails in determining the position of the object elements, then an even more computationally burdensome pixel-by-pixel comparison becomes necessary. Comparison of the z values of tiles that are spanned by a surrogate volume with the z value of the surrogate volume, as claimed, is fundamentally less computationally demanding than the method of *Aleksicy*.

In response to Examiner's argument in (c) that *Quaknine* teaches each tile in the lower resolution coarse Z-buffer covers the same screen area as the each tile in the coarse Z-buffer, please note the following:

Quaknine does not teach or suggest that "each tile in the lower resolution coarse Z-buffer covers the same screen area as each tile in the coarse Z-buffer"; there is no mention of tile system; no mention of various resolutions of the coarse Z-buffer; and no mention that the tiles in both the coarse Z-buffer and lower resolution versions of the coarse Z-buffer cover the same area. *Quaknine*, rather, speaks of subdividing the realistic rendering into portions and selected portions being rendered. The computational demand, in *Quaknine*, is reduced by not reproducing a previously rendered portion or the data invalidated by the update data. Whereas, the subject invention reduces the computational demand by using Z-buffers of various resolutions.

Accordingly, and as discussed individually per claim in the previous response, claims 1-13 are patentable over the combination of *Aleksicy* and *Quaknine*.

For the foregoing reasons, Applicant requests reconsideration and withdrawal of this rejection under 35 U.S.C. 103.

CONCLUSION

Applicant believes that this application is now in condition for allowance of all claims herein, claims 1-18, therefore, an early Notice of Allowance is respectfully requested.

Attached hereto in the subsequent pages is a marked-up version of the changes made to the specification by the current amendment. The attached pages are captioned "**Version With Markings To Show Changes Made.**"

If for any reason an insufficient fee has been paid, please charge the insufficiency to Deposit Account No. 05-0150.

If the Examiner has any questions or needs any additional information, the Examiner is invited to telephone the undersigned attorney at (650) 843-3355.

Respectfully submitted,
Angus Dorbie

Date: Nov 12, 2001

By: Fariba Sirjani
Fariba Sirjani
Attorney for Applicant
Registration No. 47,947

Squire, Sanders & Dempsey LLP
600 Hansen Way
Palo Alto, CA 94304-1043
Telephone (650) 856-6500
Facsimile (650) 843-8777

Attachment: Version With Markings To Show Changes Made

VERSION WITH MARKINGS TO SHOW CHANGES MADE

In the claims:

All of the claims have been included for completeness and for Examiner's convenience. Claims that are being amended and the new claims are identified with an asterisk (*). Claims that are being cancelled appear with a strike through the entire the claim.

None of the claims are being amended in this response. None of the claims are being cancelled. New claims 14-18 are being added.

1. (Once Amended) A method for culling occluded objects from an image being rendered into a frame buffer; the method, performed by a host processor, comprising:

constructing a coarse Z-buffer, the coarse Z-buffer subdivided into a series of tiles, each tile having an associated depth value;

updating the depth values of the coarse Z-buffer using Z information from the frame buffer; and

using the depth values to selectively discard the occluded objects from the image being rendered.

2. (Once Amended) A method as recited in claim 1, wherein updating the depth values is performed synchronously as information in the frame buffer changes.

3. (Once Amended) A method as recited in claim 1, wherein updating the depth values is performed asynchronously.

4. (Once Amended) A method as recited in claim 1, wherein using the depth values to selectively discard the occluded objects further comprises:

constructing a surrogate volume for an object; and

comparing nearest Z-value of the surrogate volume to the depth value of a tile that includes the surrogate volume.

5. (Once Amended) A method as recited in claim 4, further comprising transforming the surrogate volume from object space to eye space.

6. (Once Amended) A method as recited in claim 1, wherein using the depth values to selectively discard the occluded objects further comprises:

constructing a surrogate volume for an object; and

retrieving greatest depth value from the depth values of set of tiles that are spanned by the surrogate volume; and

comparing the nearest Z-value of the surrogate volume to the retrieved depth value.

7. (Once Amended) A method as recited in claim 6, further comprising transforming the surrogate volume from object space to eye space.

8. (Once Amended) A method as recited in claim 1, further comprising:

constructing a lower resolution coarse Z-buffer, the lower resolution coarse Z-buffer subdivided into a series of tiles, each tile having an associated depth value; and

updating the depth values of the lower resolution coarse Z-buffer using Z information from the frame buffer.

9. (Once Amended) A method as recited in claim 8, wherein each tile in the lower resolution coarse Z-buffer covers the same screen area as each tile in the coarse Z-buffer.

10. (Once Amended) A method as described in claim 8, wherein the tiles in the lower resolution coarse Z-buffer are overlapping.

11. A system, used as a host for a graphics pipeline, comprising:

a host processor executing a graphics application program, wherein the graphics application program is capable to implement:

a generation stage for creation, acquisition, and modification of information to be displayed, and organizing the information into application data structures; and

a traversal stage for traversal of the application data structures, and passing on appropriate graphics data; and

a graphics processor, communicatively coupled to the host processor, capable to implement:

a transformation stage for transformation of graphics data from object-space coordinates into eye-space coordinates, performing requested lighting operations, clipping the transformed data in clip-space, and projecting resulting coordinates into window-space;

a rasterization stage for rendering window-space primitives into a frame buffer, and performing shading calculations, texture lookups and calculations, and per-pixel operations;

a feedback loop permitting the rasterization stage to return information to the traversal stage; and

a display stage for scanning resulting pixels in frame buffer for display to a display device.

12. A system comprising:

means for constructing a coarse Z-buffer, the coarse Z-buffer subdivided into a series of tiles, each tile having an associated depth value;

means for updating the depth values of the coarse Z-buffer using Z information from a frame buffer; and

means for using the depth values to selectively discard occluded objects from an image being rendered into the frame buffer.

13. A machine-readable medium comprising instructions to a machine to:

construct a coarse Z-buffer, the coarse Z-buffer subdivided into a series of tiles, each tile having an associated depth value;

update the depth values of the coarse Z-buffer using Z information from a frame buffer; and

use the depth values to selectively discard occluded objects from an image being rendered into the frame buffer.

*14. (New) A method for early culling of occluded objects, comprising:

a) ordering all objects, the objects being included in an image being rendered, according to their distance from eye point, comprising:

i) logically dividing area of the image into a coarse Z-buffer, the coarse Z-buffer including a series of tiles, the tiles being arranged in a rectangular grid, wherein the grid may have different resolutions, and wherein each tile has an associated depth value, the depth value being a Z-buffer value farthest from the eye that is included within that tile;

ii) constructing a surrogate volume for each object of the image, wherein each surrogate volume is a three-dimensional object that is just large enough to contain the object being ordered and wherein each surrogate volume may span only one tile of an appropriate resolution;

iii) determining a depth value of the surrogate volume that is nearest to eye of a viewer;

iv) determining a depth value of the one tile that includes the surrogate volume;

v) comparing the depth value of the surrogate volume versus the depth value of the tile including the surrogate volume;

b) culling the objects whose surrogate volume has a depth value farther from the eye than the depth value of the tile, including the surrogate volume, after a single comparison; and

c) rendering the objects whose surrogate volume has a depth value closer to the eye than the depth value of the tile, including the surrogate volume, or equidistant to the eye with the depth value of the tile including the surrogate volume.

*15. (New) The method of claim 14, wherein a surrogate volume may span several tiles and further comprising:

comparing the depth value of the surrogate volume with each of the spanning tiles; and

culling the objects whose surrogate volume has a depth value farther from the eye than the depth value of the tiles including the surrogate volume; and

rendering the objects whose surrogate volume has a depth value closer to the eye than the depth value of at least one of the tiles including the surrogate volume or is equidistant to the eye with at least one of the tiles including the surrogate volume.

. (New) The method of claim 14, further comprising:

subdividing the objects that are not occluded into smaller objects; and

determining if the smaller objects are occluded.

*17. (New) The method of claim 14, wherein

each coarse Z-buffer is replicated one or more times at different resolutions,

each separate coarse Z-buffer spans the image using a different resolution,

the number of tiles in the coarse Z-buffers of various resolutions remains constant,

for lower resolution coarse Z-buffers, each tile covers a larger area of the image,

for lower resolution coarse Z-buffers, the tiles overlap one another,

center points of successive resolutions of tiles of the coarse Z-buffers are offset from the center points of preceding resolutions of tiles,

lower resolution tiles of the coarse Z-buffers split the image between tiles with overlap,

a higher resolution coarse Z-buffer splits the image between tiles with no overlap, and

a host processor is allowed to select a resolution that corresponds to a size of any given object.

*18. (New) A system comprising:

a memory for storing depth values, wherein the depth values are derived by:

a) ordering all objects, the objects being included in an image being rendered, according to their distance from eye point, comprising:

i) logically dividing area of the image into a coarse Z-buffer, the coarse Z-buffer including a series of tiles, the tiles being arranged in a rectangular grid, wherein the grid may have different resolutions, and wherein each tile has an associated depth value, the depth value being a Z-buffer value farthest from the eye that is included within that tile;

ii) constructing a surrogate volume for each object of the image, wherein each surrogate volume is a three-dimensional object that is just large enough to contain the object being ordered and wherein each surrogate volume may span only one tile of an appropriate resolution;

iii) determining a depth value of the surrogate volume that is nearest to eye of a viewer;

iv) determining a depth value of the one tile that includes the surrogate volume;

a host processor for:

determining which objects are occluded by comparing the depth value of the surrogate volume versus the depth value of the tile including the surrogate volume,

generating the surrogate volumes for the objects being processed, and

transforming the surrogate volumes from object space to eye space; and

a graphics processor for rendering the objects that are not occluded.